Remarks/Arguments

This Amendment is in response to the Office Action mailed 27 October 2003 (27.10.2003). In this Office Action, the Examiner objected to the title of the invention as being not descriptive. Additionally, the Examiner rejected claims 1-4 and 6-8 under 35 USC 103(a) as being unpatentable over Cottle et al. (US Pat. No. 6,263,396) in view of Applicant's Admitted Prior Art. The Examiner found allowable subject matter in claim 5, if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

1. Title Objection

The Examiner objected to the title as not being descriptive of the invention. Applicant is amending the title to better reflect the claimed invention. Therefore, Applicant respectfully requests that the Examiner withdraw the objection to the title.

2. Summary of Current Claims

Claims 1-23 remain in this application. Claims 1 and 7 have been amended. Claims 6 and 8 have been canceled. In addition, claims 9-23 have been added. Claims 9-13 are system claims that are commensurate in scope with apparatus claims 1-5, as amended herein. Likewise, claims 14-18 are method of manufacture claims, and claims 19-23 are method of use claims that are commensurate in scope with apparatus claims 1-5, as amended herein. Since the new claims are commensurate in scope with the apparatus claims already presented and amended herein, Applicant believes that a new search is unnecessary due to the addition of claims.

The Examiner has acknowledged that claim 5 is directed to allowable subject matter and would be allowed if rewritten in independent form including all of the limitations of the base claim and any intervening claim. However, rather than rewriting claim 5, Applicant is amending independent claim 1, from which claim 5 depends, to include limitations not found in the prior art.

3. Claim Rejections under 35 USC 103(a)

The Examiner rejected claims 1-4 and 6-8 under 35 USC 103(a) as being unpatentable over Cottle et al. (US Pat. No. 6,263,396) in view of Applicant's Admitted Prior Art. In response to the Office Action, Applicant is amending the claims to further clarify the claimed invention, to make the claims more readable, and to conform the claims to the disclosure. Applicant believes that a new search is unnecessary as these amendments narrow the scope of the claimed invention. Further, Applicant believes that these amendments place the pending claims in immediate condition for allowance or appeal.

The Examiner asserts, and Applicant agrees, that Cottle discloses a programmable interrupt controller on a single IC that uses an ARM core that switches between ARM (32-bit) and THUMB (16-bit) instruction mode. Cottle is fully aware of the efficiency differences between ARM and THUMB execution, and provides a reasonably cogent description of the ARM and THUMB program execution modes at col. 17, lines 1-23. The instant application discloses the existence of the "BX instruction" that toggles the ARM core between 32-bit ARM execution and 16-bit

THUMB execution. The Examiner has designated Applicant's description of this instruction as "Applicant's Admitted Prior Art" (AAPA).

However, Cottle's invention is not concerned with the ARM and THUMB program execution modes of the ARM core, nor the exploitation of the two different program modes for processing interrupts. Cottle's invention and disclosure focuses on using a programmable interrupt mask and a programmable interrupt vector table to enable the masking of interrupts, the prioritization of interrupts, and the servicing of multiple interrupts (ie., interrupting a lower-priority interrupt to service a higher-priority interrupt.) Cottle discloses that his programmable interrupt vector table enables the dynamic association of interrupt service routines with specific interrupts (Col. 5, lines 41-50), but Cottle does not disclose specifically using THUMB programming code for his interrupt service routines to save program code space. Likewise, Cottle does not disclose using a shared ARM-encoded interrupt service routine preamble to put the ARM core into THUMB mode to execute a THUMB-encoded interrupt service routine. Indeed, Cottle teaches away from exploiting the core's THUMB mode by purposefully negating the core's interrupt-triggered hardware switch, teaching us instead that for the purposes of his disclosure, "ARM and Thumb are used interchangeably herein." Col. 17, lines 14-15. In contrast, one of the specific objects of the present invention is to save program code space by purposefully negating the core's switch to ARM mode to service interrupts (using the interrupt service routine preamble of the present invention) and further, to efficiently accomplish that negation for all interrupt service routines using a single, shared preamble.

As disclosed in the instant application, the ARM core switches, by design, to 32-bit ARM processing whenever an interrupt is received. Application at p. 2, lines 19-21. This feature of the ARM processor would ordinarily require that interrupt service routines be encoded in 32-bit ARM code. Alternatively, as disclosed in the instant application, if program code space is tight and system designers prefer to use space-saving THUMB program code for their interrupt service routines, they must first include a few ARM-encoded instructions to put the processor into THUMB mode so that the THUMB-encoded interrupt service routine can be executed. Page 2, lines 22-24. Therefore, in a program space-constrained hardware environment, there is needless duplication since every interrupt service routine will begin the same way: with the ARM-encoded instructions that switch the processor to THUMB mode. The present invention eliminates this wasteful duplication by using a single interrupt service routine preamble that provides this function and that can be shared amongst all interrupt service routines, thus eliminating the need for THUMB-encoded interrupt service routines to include these extra instructions.

Accordingly, upon receiving an interrupt, the interrupt controller of the present invention first directs execution to an interrupt service routine preamble encoded in 32-bit ARM code that switches the ARM core back into 16-bit THUMB mode. Thereafter, an interrupt service routine encoded in 16-bit THUMB instructions can be executed. The use of a single ARM-encoded interrupt service routine preamble that is shared by all the interrupt service routines enables the interrupt service routines to be encoded in space-saving THUMB code and eliminates the need for redundant ARM code to switch

the core to THUMB mode within every individual interrupt service routine. In contrast, Cottle does not disclose any specific encoding requirement for his interrupt service routines, and presumably, Cottle's interrupt service routines are encoded in ARM code since Cottle does not describe any specific effort to switch the core from ARM to THUMB after an interrupt is received.

The present amendments clarify this function of the interrupt service routine preamble and clarify that the claimed preamble comprises ARM-coded instructions that cause the processor core to switch to the THUMB mode. In claim 1 as amended, ARM coded instructions are generically claimed as a "first program execution stream" and THUMB coded instructions are generically claimed as a "second program execution stream, wherein said second program execution stream is more economical with program code space than said first program execution stream." Claim 1, as amended, now claims the interrupt service routine preamble as follows:

> "an interrupt service routine preamble shared amongst said plurality of interrupt service routines, said interrupt service routine preamble is coded in said first program execution stream to cause a hardware switch to said second program execution stream for executing one of said plurality of interrupt service routines."

Likewise, Claim 7, as amended, now claims the interrupt service routine preamble as follows:

> "an interrupt service routine preamble shared amongst said plurality of interrupt service routines, said interrupt service routine preamble is coded in ARM program code to cause a switch to THUMB program execution for executing one of said plurality of interrupt service routines."

In the Office Action, the Examiner asserted that Cottle teaches an interrupt service routine preamble, citing Col. 14, lines 47-56, that is shared amongst a plurality of interrupt service routines, citing Col. 17, lines 35-55. However, the first reference is a description of part of the architecture of Cottle's system-on-a-chip decoder circuit 200 (shown in FIG. 1B) that forms a portion of Cottle's overall communications system 100. In the lines referenced by the Examiner, Cottle describes that the circuit's three interrupt lines are each connected to a different IRQ. When the ARM's interrupt handler begins servicing one of these IRQs, it asserts an interrupt acknowledge signal, EXTACK. After the IRQ is serviced, the ARM resets the EXTACK signal. Therefore, the lines cited by the Examiner as teaching an interrupt service routine preamble relate to the assertion and deassertion of interrupt acknowledge signals in the hardware; they clearly do not teach a software interrupt service routine preamble "coded in a specific program execution stream [i.e., ARM code] to cause a hardware switch to a second program execution stream, where said second program execution stream is more economical with program code space than said first program execution stream [i.e., THUMB code]" as the interrupt service routine preamble element is currently claimed.

Similarly, Col. 17, lines 35-55 do not teach an interrupt service routine preamble, having the above-described limitations, that is shared amongst a plurality of interrupt service routines. Instead, these lines provide a generic description of the kinds of interrupts the ARM receives and the kinds of services that are required in response to these interrupts. They also provide a basic description of the higher-level functionality of Cottle's traffic controller module 300 ("manages interrupt requests...DMA transfers..[and] the extension bus....[p]rovides memory access protection and manages

the data flow between processors and memories [citing examples].") These lines do not relate to the shared interrupt service routine preamble as currently claimed.

Given the failure of Cottle to disclose an ARM-encoded interrupt service routine preamble that switches the ARM core to THUMB mode and that is shared amongst interrupt service routines, a prima facie case of obviousness, using Cottle and the BX instruction as 103(a) references, cannot be established. Among other things, MPEP 2143 requires that the combination of the cited references teach or suggest all of the claim limitations and that some motivation or suggestion to combine the references must exist. Cottle does not teach an interrupt service routine preamble, and the only teaching that Cottle does provide that relates to interrupt service routines is the ability to dynamically associate interrupt service routines with specific interrupts. Cottle certainly does not teach exploiting the ARM core's capability to execute instructions in THUMB mode to save program space, and teaches away from doing so since, in Cottle's disclosure, ARM and THUMB "are used interchangeably" Col. 17 lines 14-15. Since, for Cottle's purposes, ARM and THUMB are "interchangeable," there would be no need or suggestion for a practitioner of Cottle's invention to use the BX instruction at all.

Consequently, the combination of Cottle's teaching and the existence of the BX instruction used by ARM programmers to toggle between 32-bit ARM mode execution and 16-bit THUMB mode execution does not establish the above-claimed interrupt service routine preamble, having the above-claimed functional and structural characteristics.

In sum, Cottle's disclosure, combined with the existence of the BX instruction, does not teach or suggest all of the claim limitations of the currently claimed invention.

Additionally, Cottle fails to provide a suggestion or motivation to one skilled in the art to use an interrupt service routine preamble to exploit the ARM core's THUMB mode to process interrupts to save program code space, and moreover, to share that interrupt service routine preamble amongst a plurality of THUMB-encoded interrupt service routines. Therefore, Applicant respectfully requests that the Examiner withdraw the rejections to claims 1-4 and 6-8 under 35 USC 103(a) as being unpatentable over Cottle et al. (US Pat. No. 6,263,396) in view of Applicant's Admitted Prior Art.

4. Summary

In view of the above, Applicant believes that each of the presently pending claims is in immediate condition for allowance or appeal. Accordingly, Applicant respectfully requests that the Examiner withdraw the outstanding objections and rejections of the claims and issue a timely Notice of Allowance in this case.

Respectfully submitted,

Date: January 27, 2004

Karen S. Wright

Karen S. Wright
Reg. No. 45,240

Matthew J. Booth
Karen S. Wright
Booth & Wright, L.L.P.
PO Box 50010
Austin, Texas 78763-0010
Telephone: (512) 474-8488
Facsimile: (512) 474-7996
matthew.booth@boothlaw.com
karen.wright@boothlaw.com
http://www.boothlaw.com/